

Air:bit

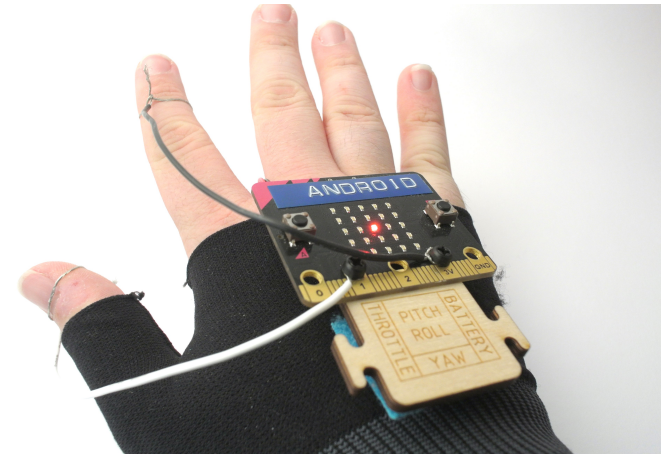
Smart hanske



Ferdig hanske

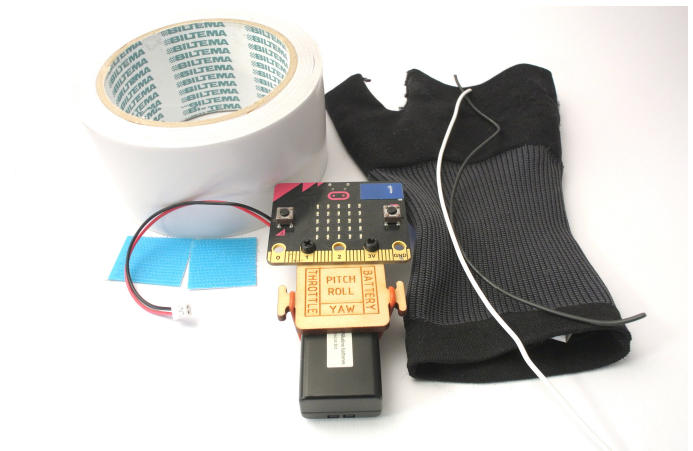
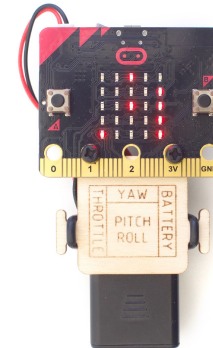
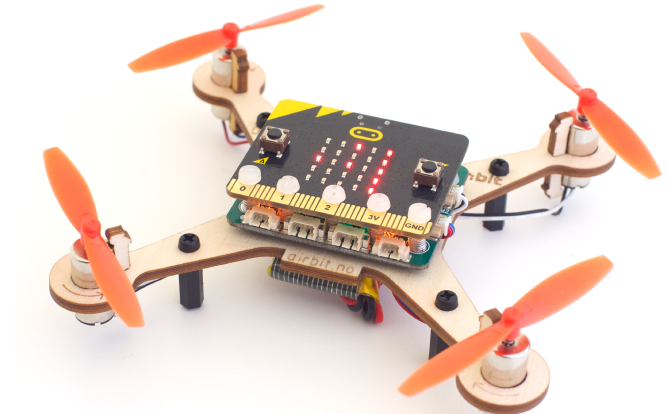
Her ser du den ferdige smart hansken!

Med hjelp av denne hansken kan du styre Air:bit ved hjelp av håndbevegelse. Du bruker fortsatt A og B for mer eller mindre throttle, men ved hjelp av fingersensorene så kan du også gi en boost for mer throttle, ved å klemme sammen fingrene som du har kablene rundt. Når du ikke lenger klemmer kablene sammen, vil throttle gå ned til det nivået du hadde, før du klemte fingrene sammen



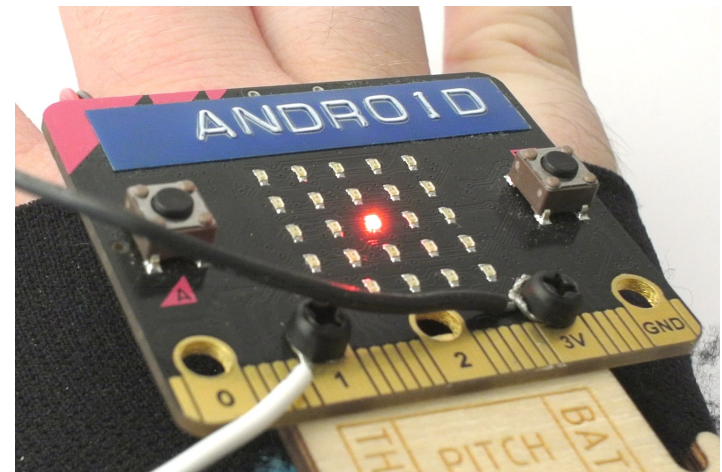
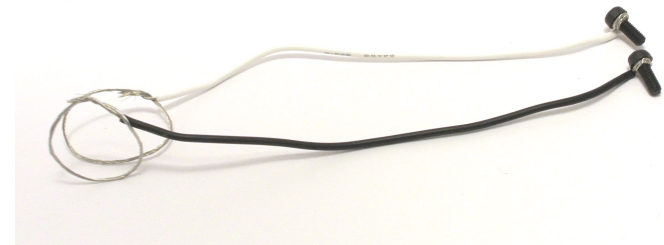
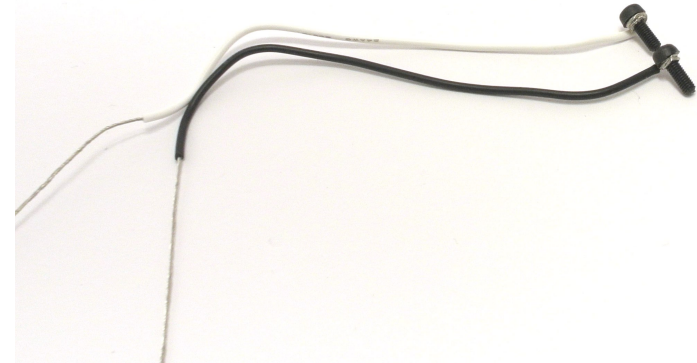
Deler

- 1 stk ferdig bygget Air:bit
- 1 stk håndkontroller
- Borrelås/dobbeltsidig teip
- En hanske som du kan klippe fingrene av
- To kabler på ca. 20 cm.



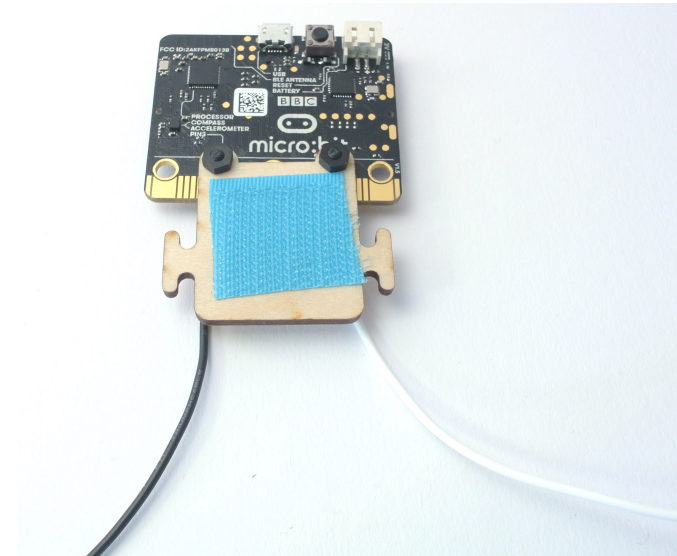
Montering

- Hanske (som du kan klippe fingrene av), og kablene som skal avisoleres, ca 6-8 cm i den ene enden som skal rundt fingrene, og ca 2-3 cm som skal skrus fast med skruen
- Tvinn kablene rundt skruen, slik at de sitter godt fast
- Tvinn så kablene rundt tommelen, og pekefingeren din, slik at kablene får en rund ende
- Kablene skal deretter skrus fast til Pin 1 og 3V

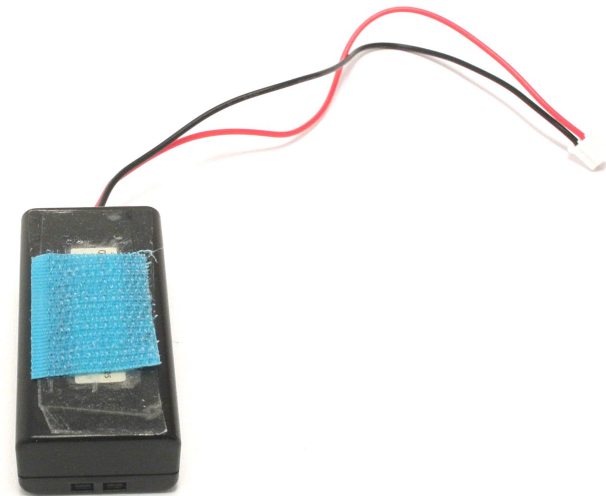


Montering

- Fest en borrelås bit til baksiden av batteriholder braketten



- Fest en borrelås bit til baksiden av batteriholderen



Montering

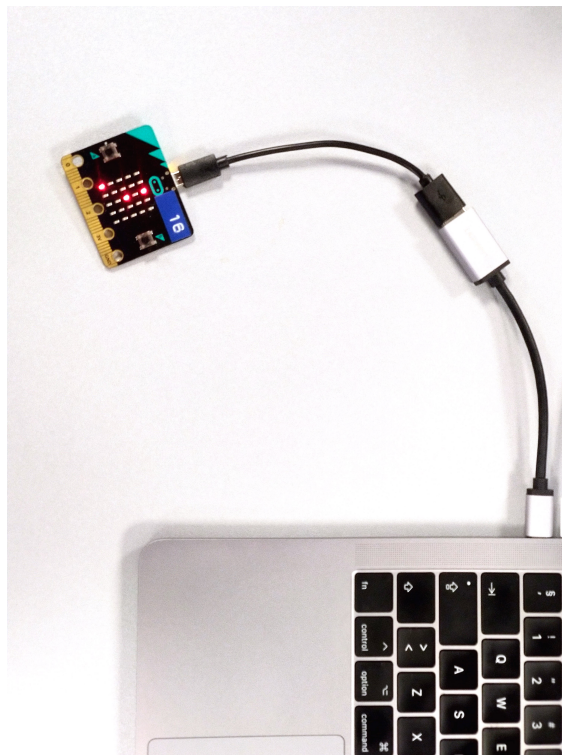
- Fest motparten til borrelåsen på forsiden, og baksiden av hansken



Micro:bit kontrollen, skal på oversiden av hansken, batteriboksen skal på undersiden



Koble Micro:bit til Mac/PC

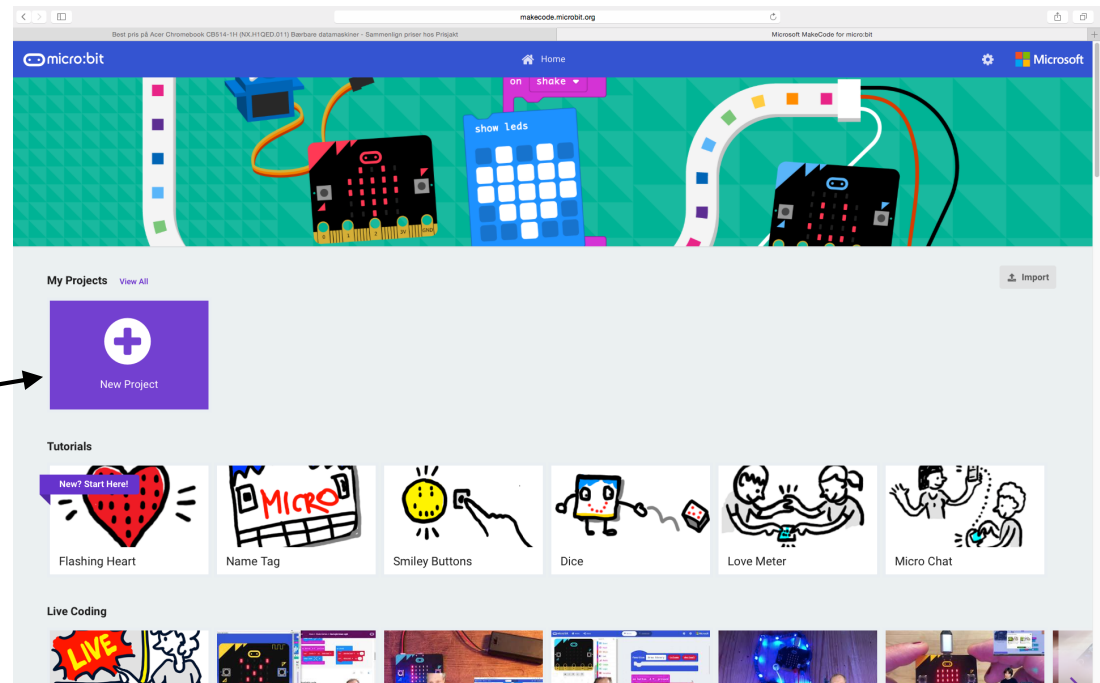


MakeCode

- Slik ser nettsiden vi skal bruke til testing av fingersensoren, men også kodingen

<https://makecode.microbit.org>

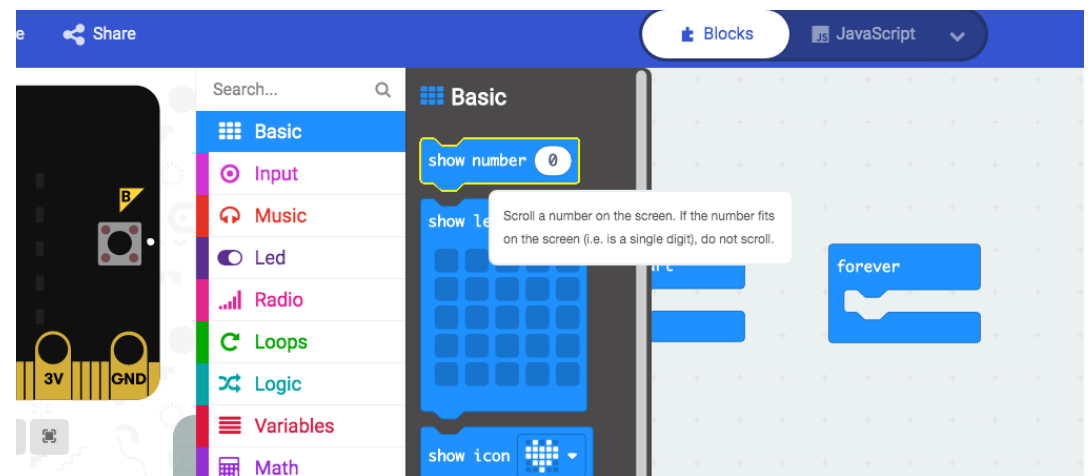
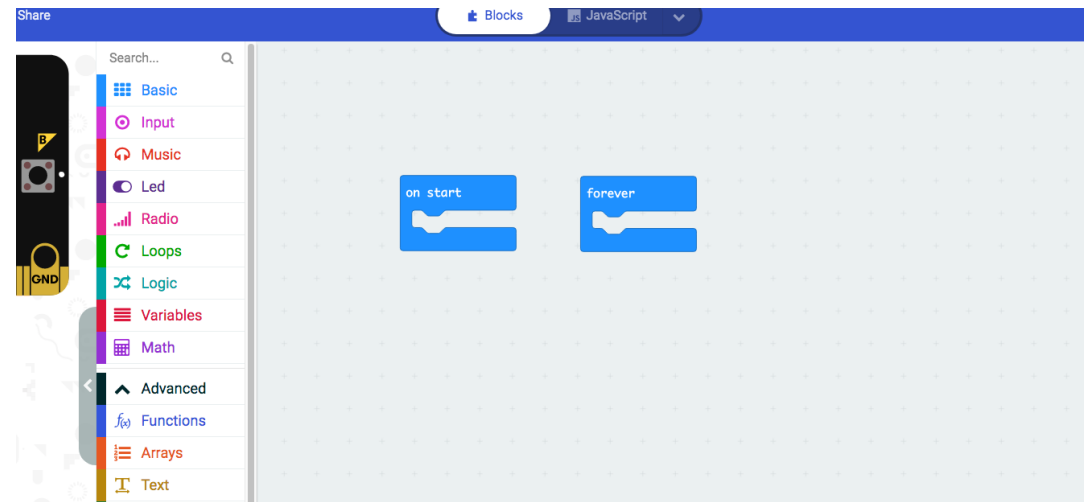
- Før hvert nytt prosjekt, må vi klikke på "New Project"



Testing av fingersensor

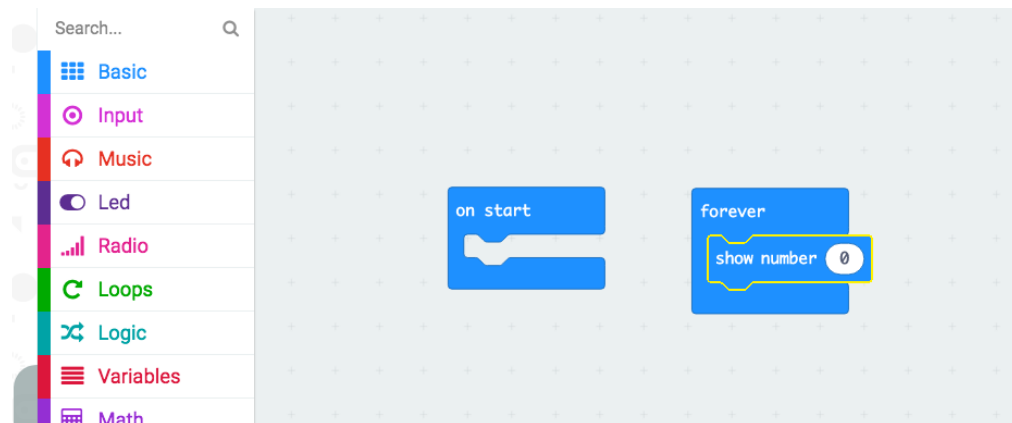
- Først må vi finne ut av hva spenningen mellom kablene er, når de ikke er klempt sammen, og når de er klempt sammen.
For å finne ut av spenningen mellom kablene, så trenger vi noen kode-blokker.
Se først etter show number 0, i Basic blokken

Vi måler spenningen, for å finne ut av den laveste, og høyeste verdien, slik at vi kan legge korrekt verdi inn i koden. Hvis du legger inn en verdi som ikke er korrekt, vil ikke fingersensoren fungere.

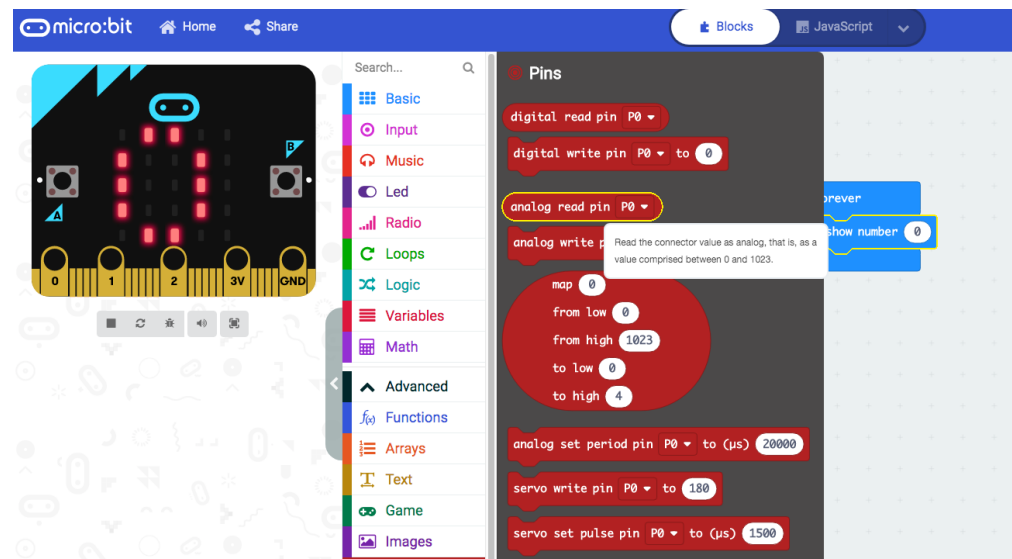


Testing av fingersensor

- Plasser "show number" blokken i forever blokken.



- Nå trenger vi en blokk fra Pins, se etter "analog read pin P0"



Testing av fingersensor

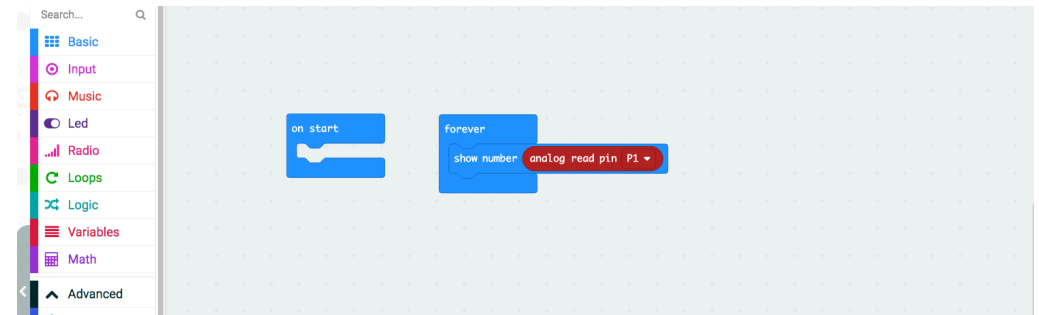
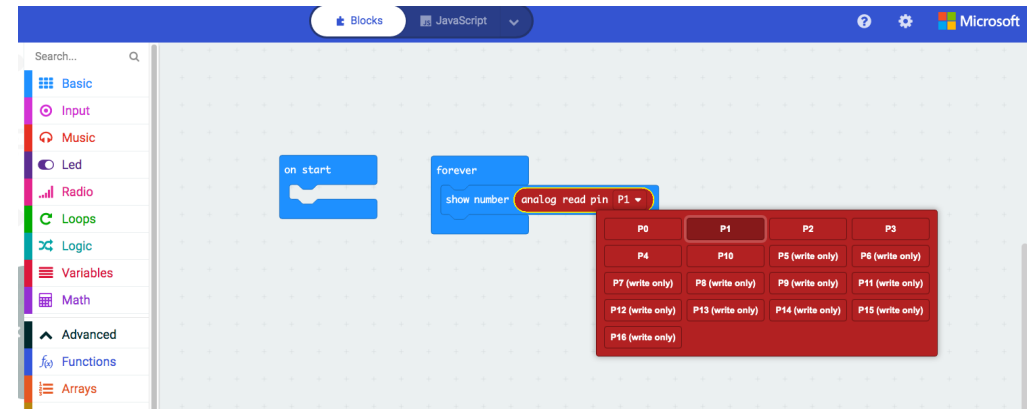
- Plasser “analog read pin P0” i “show number” blokken
- Endre P0 til P1

Dette fordi vi vil finne ut av hva spenningen mellom pin 1, og 3V er

Trykk på download, slik at koden lastes over til Micro:bit kontrollen, og vi er klar for testing

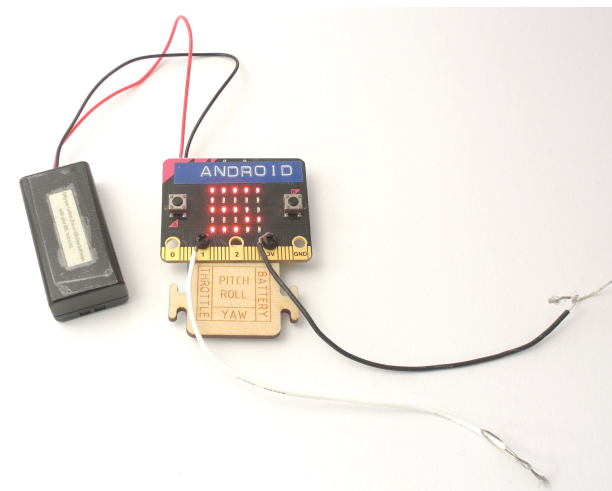
Observer forskjellen mellom når du trykker, og ikke trykker.

Ett trykk vil som regel gi en verdi over 1020, og når man ikke trykker fingersensoren sammen, er verdien som regel under 1000

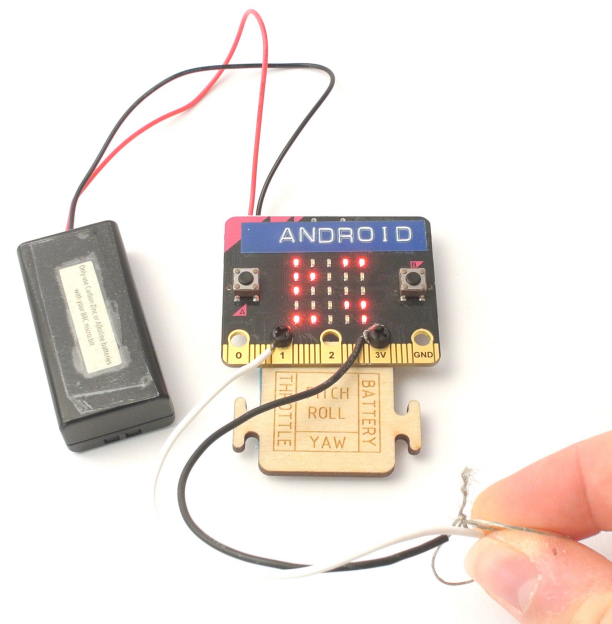


Testing av fingersensor

- På dette bildet ser du at noen tall går over skjermen, hele dette tallet, f.eks. 995, er verdien (spenningen), når kablene ikke berører hverandre.



- På det siste bildet ser du også tall som går over skjermen, hele dette tallet, f.eks. 1023, er verdien (spenningen), når kablene berører hverandre. Vi må velge en verdi som er høyere enn 995, men lavere enn 1023. Vi prøver oss på 1010.



Koding

- Klikk deg inn på denne nettsiden, <https://www.makekit.no/airbit-docs>
- Der skal du laste ned koden til Air:bit sin håndkontroll, kalt "Code for transmitter(readymade)"
- Når du har lastet ned koden, så er du klar for neste steg

Air:bit DOCS

Instructions:

Air:bit

[Norwegian instructions: Bygging + koding + Fasit](#)
[English instructions: Assembly, coding and flight](#)
[Code for transmitter\(readymade\)](#)
[Code for drone](#)
[Code for monitoring drone+transmitter](#)

Wifi-camera

[Instruction Manual Air:bit Wifi Camera \(ENG\)](#)
[Instruksjonsmanual Air:bit Wifi Kamera \(NOR\)](#)

iOS app

[iOS: Guide \(ENG\)](#)
[iOS: Guide \(NOR\)](#)
[iOS: code for drone](#)

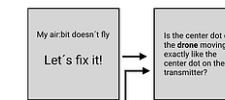
Android app

[Android: Setup guide](#)
[Android: App for phone/tablet](#)
Note: App asks for location because of bluetooth. Please allow "when using app". No position is recorded.
[Android: Code for drone](#)

Video Tutorials



Video Tutorials



[Error searching diagram](#)

Koding

- Bruk koden i MakeCode, på denne nettsiden <https://makecode.microbit.org>
- Nå skal vi laste inn koden, slik at vi kan modifisere den
- Det gjør du ved å enten ha startside til MakeCode oppe, finn mappen hvor filen ble lastet ned til (som oftest downloads/nedlastninger), hold inne venstre-klikk på filen du skal bruke, for så å dra musen over til startside til MakeCode, å slippe venstre-klikk. Da vil koden lastes inn av seg selv
Eller ved å trykke på "import" knappen til høyre på siden, trykk på "import file...", klikk på "choose file", finn mappen der koden ble lastet ned til, dobbeltklikk, og siden vil laste inn koden

 Import

Import


Import File...
Open files from your computer


Import URL...
Open a shared project URL or GitHub repo


Your GitHub Repo...
Clone or create your own GitHub repository

Open .mkcd or .hex file

Select a .mkcd or .hex file to open.

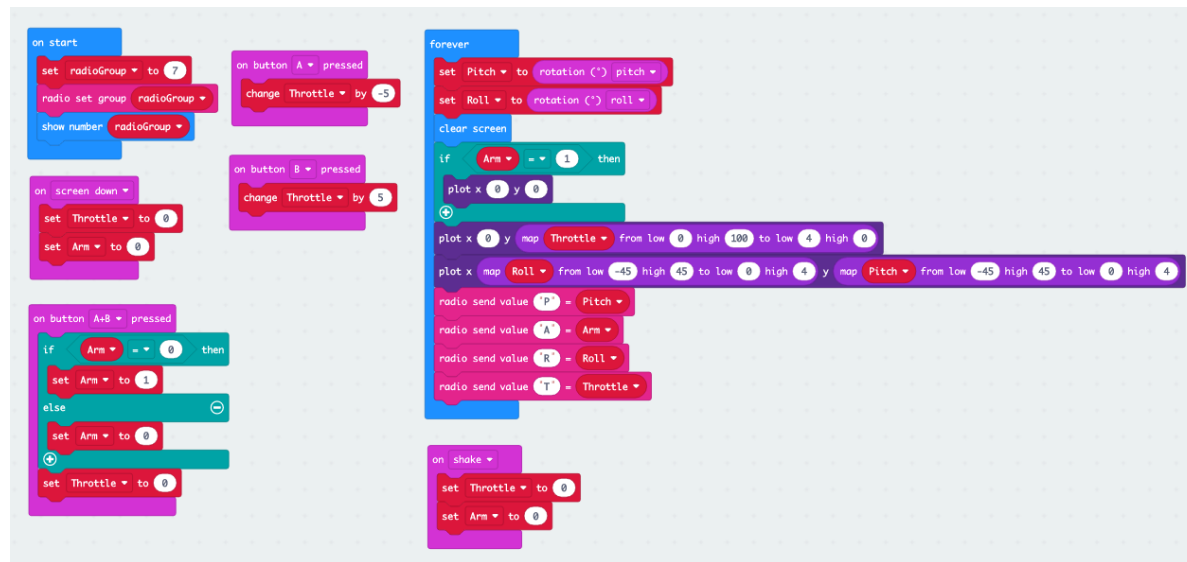
Choose File

No file chosen

You can import files by dragging and dropping them anywhere in the editor!

Go ahead!

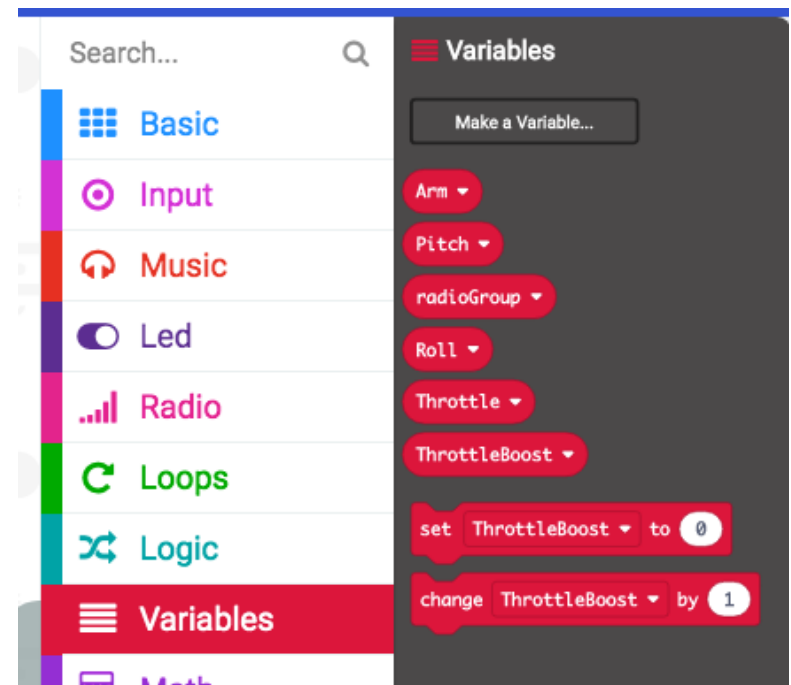
Koding



- Slik ser den ferdige koden for Air:bit Drone, ut, og vi skal bygge videre på denne koden, i Forever blokken, og kan lastes ned fra <https://www.makekit.no/airbit-docs>

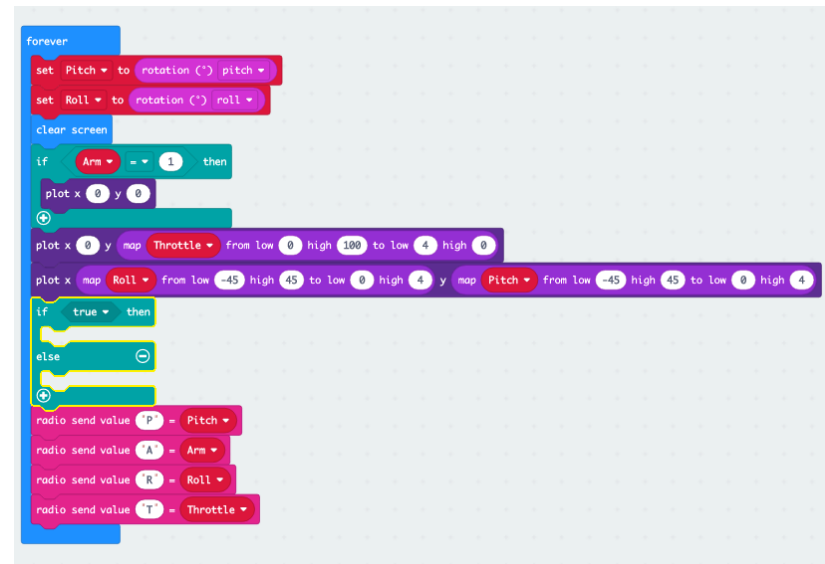
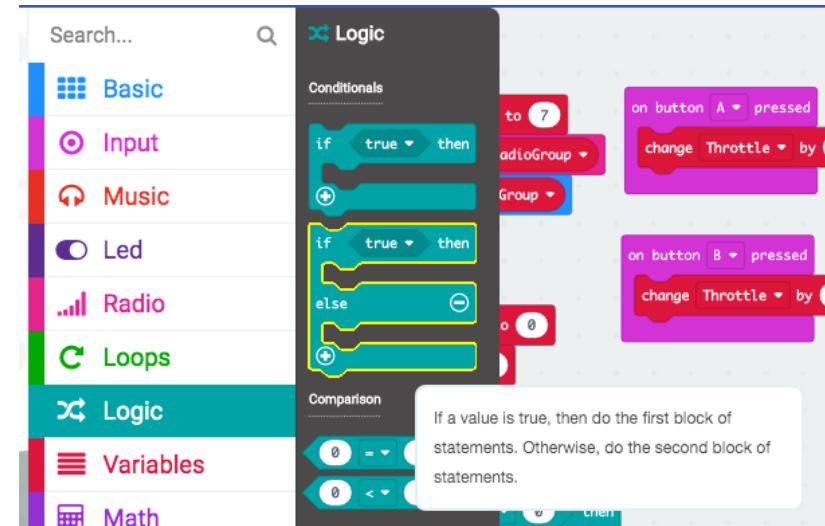
Variabler

- Her ser du variablene som er lagt inn fra før. Vi trenger kun en ekstra: "ThrottleBoost"
- Det gjør du ved å trykke på Make a Variable, i Variables, så skriver du inn navnet på variabelen, i dette tilfellet "ThrottleBoost", og trykk ok



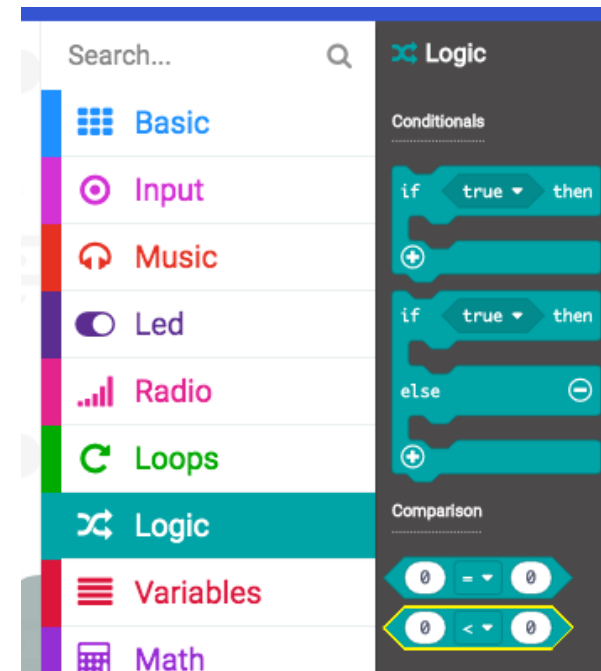
Logikk

- Så trenger vi logikk, det finner du i Logic blokken, og vi ser etter “If true then else” blokken
- Plasser “If true then else” Logic, i Forever loopen, under plot, og over blokkene med “radio send value”

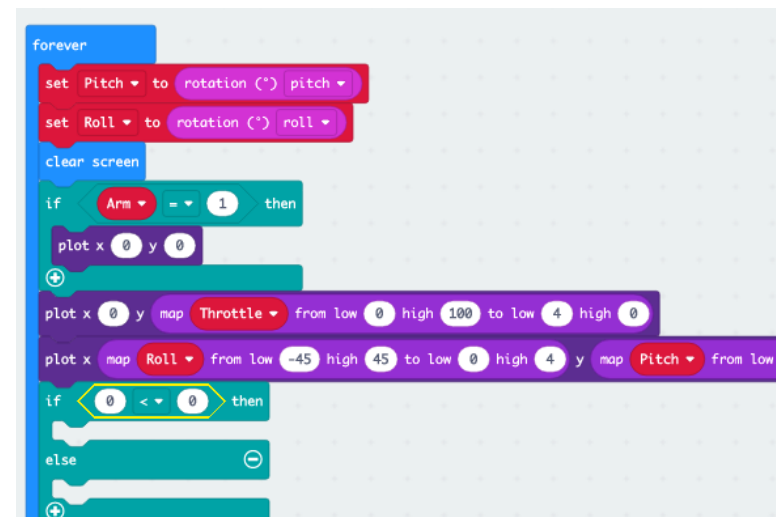


Logikk

- Så trenger vi en “comparison”, den ligger også under Logic, det er blokken opplyst i gult vi trenger

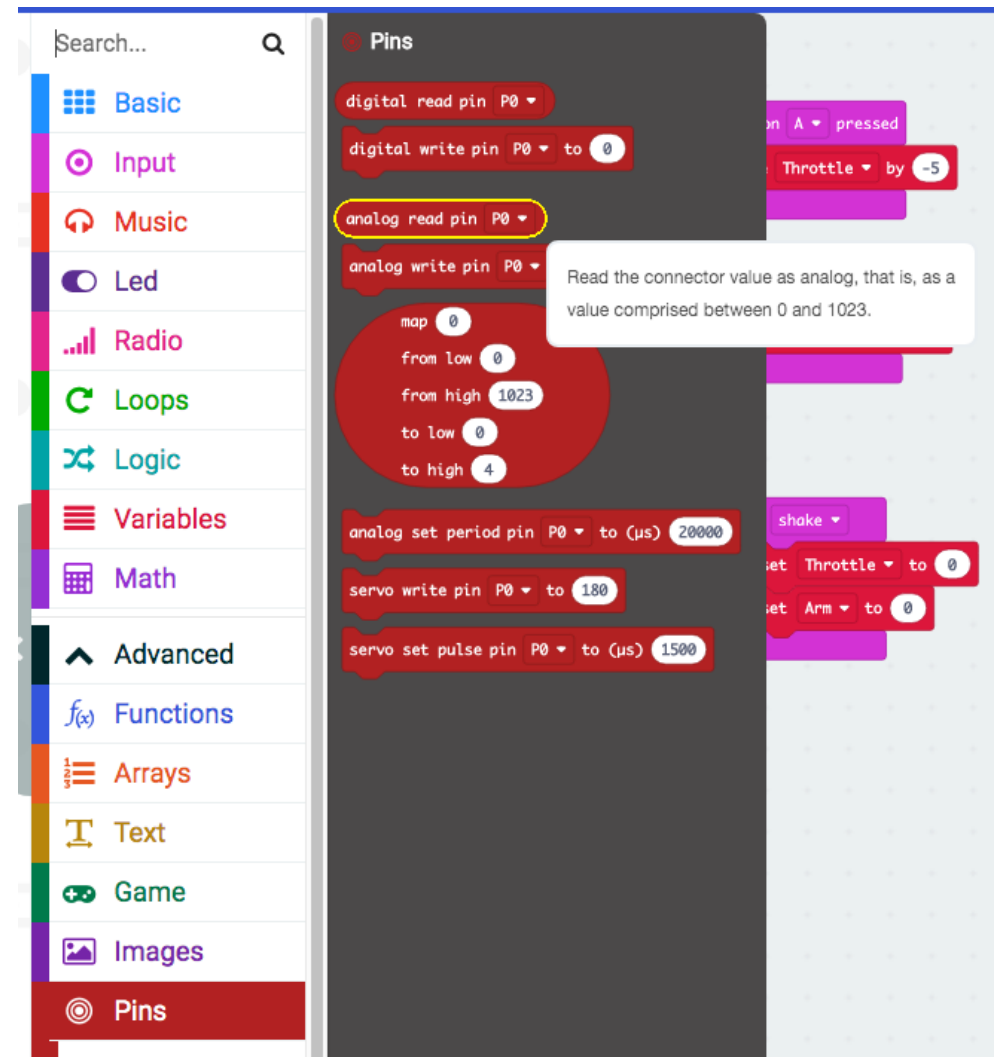


- Plasser “comparison” der det står true, så vil du se at true forsvinner, og comparison ligger der i stedetfor



Logikk og Pins

- Nå trenger vi Pins
Vi ser etter “Analog read pin P0” blokken



Pins

- Plasser “analog read pin P0” blokken i comparison blokken, til høyre

```
forever
  set Pitch to rotation of pitch
  set Roll to rotation of roll
  clear screen
  if Arm == 1 then
    plot x 0 y 0
    plot x 0 y map Throttle from low 0 high 100 to low 0 high 4
    plot x map Roll from low -45 high 45 to low 0 high 4 y map Pitch from low -45 high 45 to low 0 high 4
    if analog read pin P0 < 0 then
    else
    radio send value P = Pitch
    radio send value A = Arm
    radio send value R = Roll
    radio send value T = Throttle
```

- Skift slik at det står P1, og ikke P0
- Skriv inn verdien 1010 til venstre

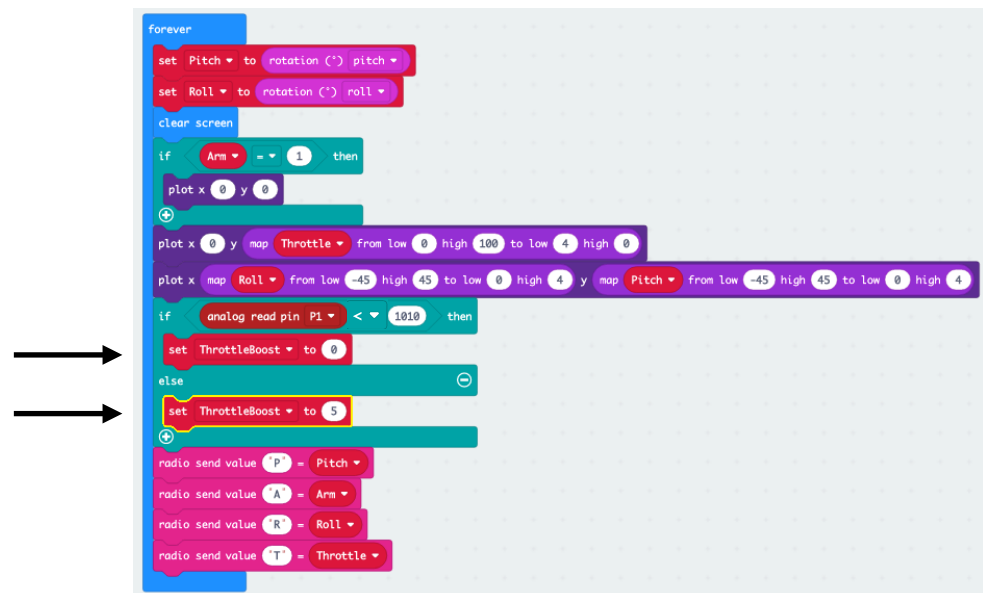
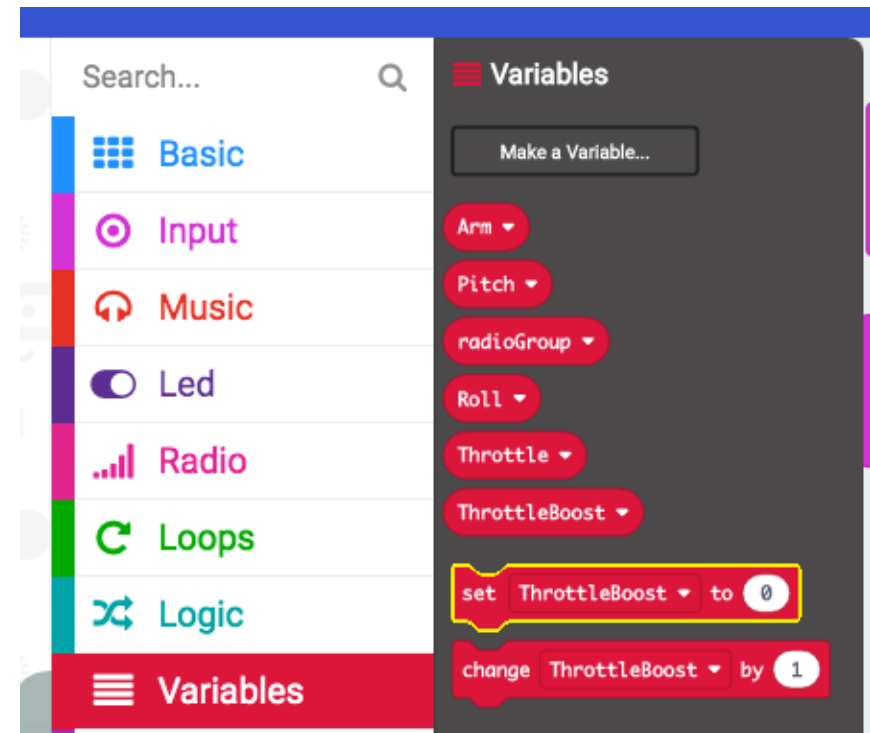
Dette er verdien vi fant ut i testing av fingersensoren
Verdien 1010 vil variere etter hvor svett du er i hånden,
så vær obs på det



```
forever
  set Pitch to rotation of pitch
  set Roll to rotation of roll
  clear screen
  if Arm == 1 then
    plot x 0 y 0
    plot x 0 y map Throttle from low 0 high 100 to low 0 high 4
    plot x map Roll from low -45 high 45 to low 0 high 4 y map Pitch from low -45 high 45 to low 0 high 4
    if analog read pin P1 < 1010 then
    else
    radio send value P = Pitch
    radio send value A = Arm
    radio send value R = Roll
    radio send value T = Throttle
```

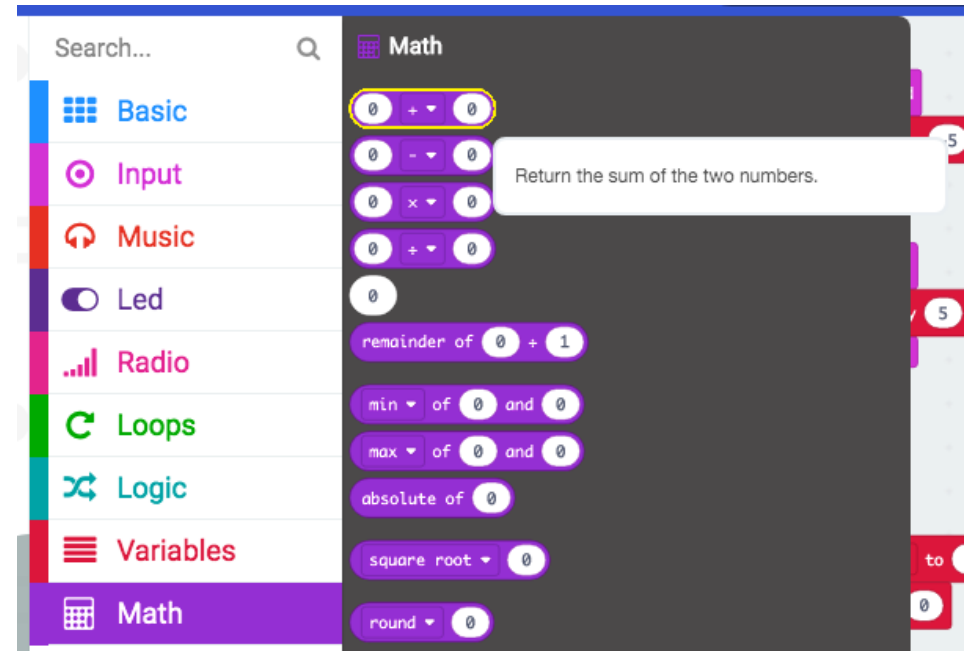
Variabler

- Klikk på variables for å finne “ThrottleBoost” variabelen
- Plasser set “ThrottleBoost to 0” rett under analog read pin P1
- Plasser set “ThrottleBoost to 0”, rett under else
- Endre verdi på den nederste til 5

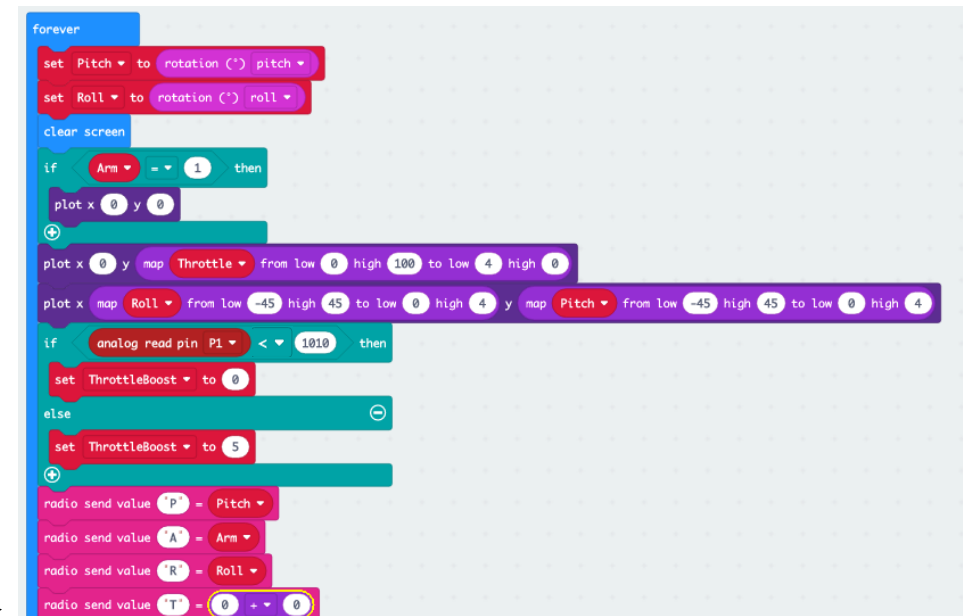


Matte

- Nå trenger vi en matte-blokk med +

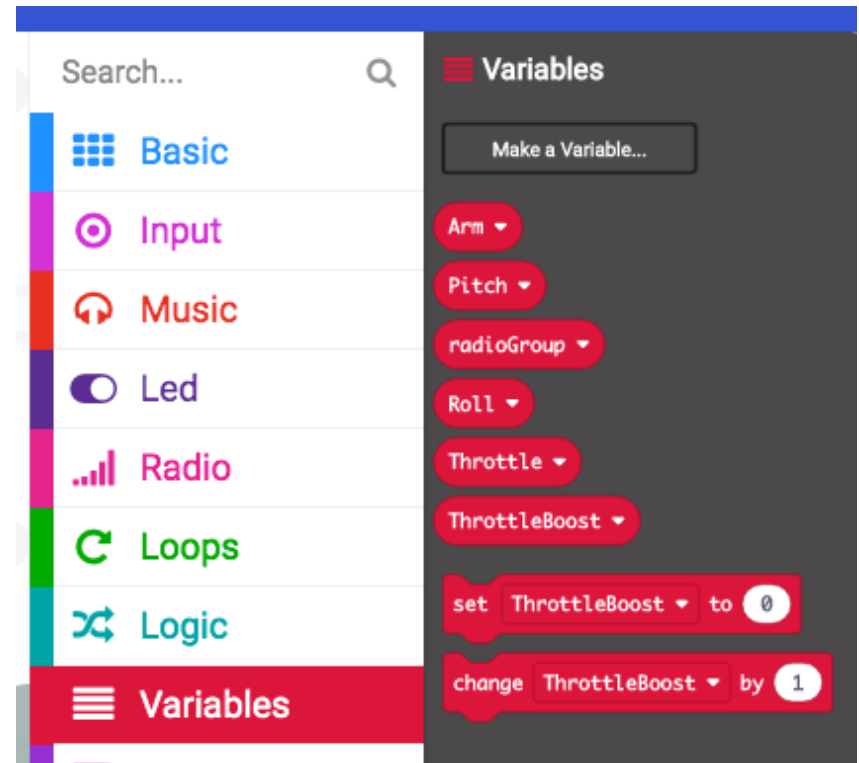


- Plasser den der tallet 0 vises, i radio send value "T" - blokken

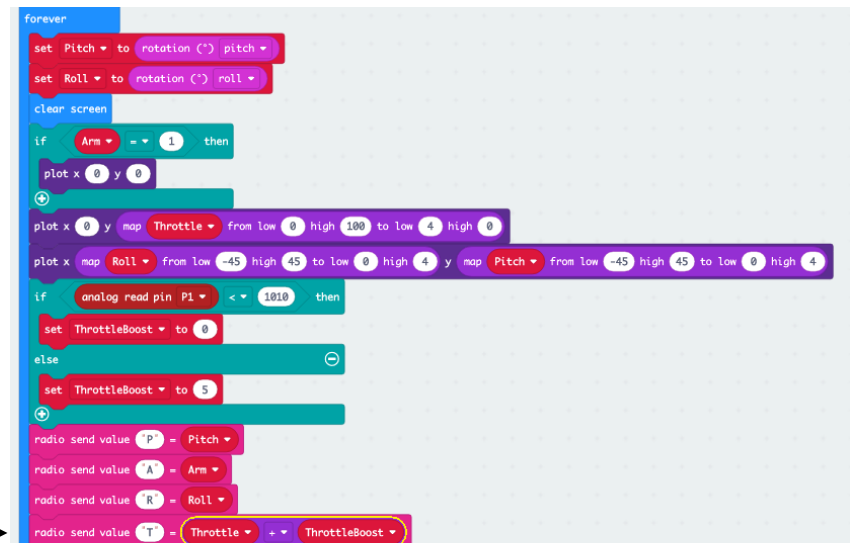


Variabler

- Nå trenger vi to variabler, en "Throttle", og en "ThrottleBoost"



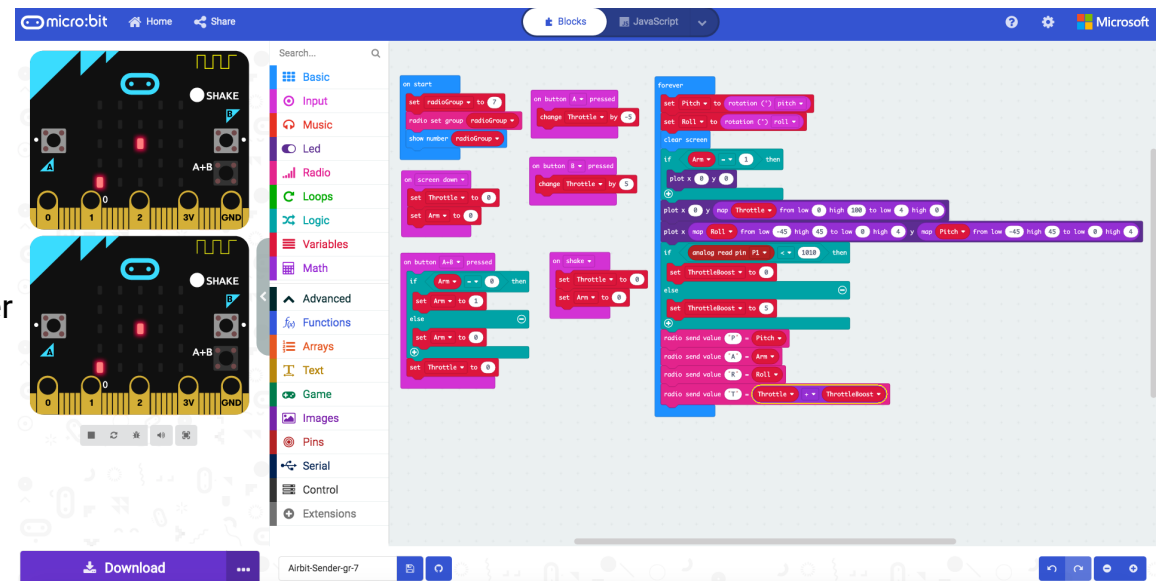
- Plasser "Throttle" i Math variabelen vi nettopp la inn, i den første 0 rubrikken, i radio send value "T" - blokken
- Plasser "ThrottleBoost" i den siste rubrikken i Math blokken



Variabler - og Download

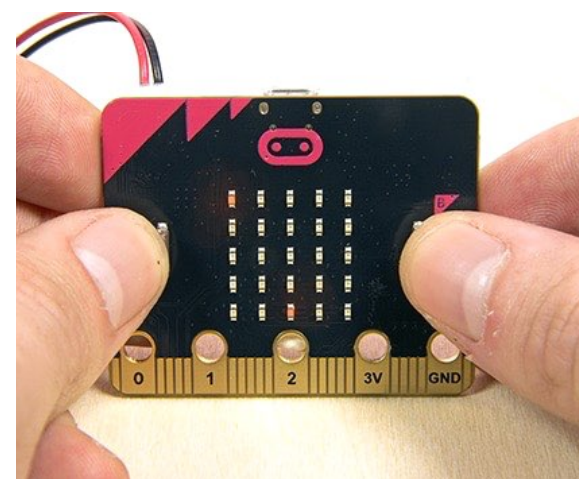
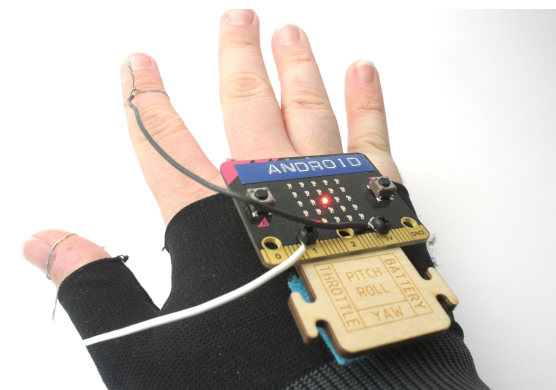
- Nå er koden komplett, og vi er klar for å koble til Micro:biten til PCen, og laste ned koden.

Hvis fingersensoren ikke responderer, kan det være fordi variabelen som er lagt inn i "analog read pin P1", er feil. Da må du gå tilbake til testing av fingersensoren, å finne korrekt verdi.



Hvordan bruke hansken

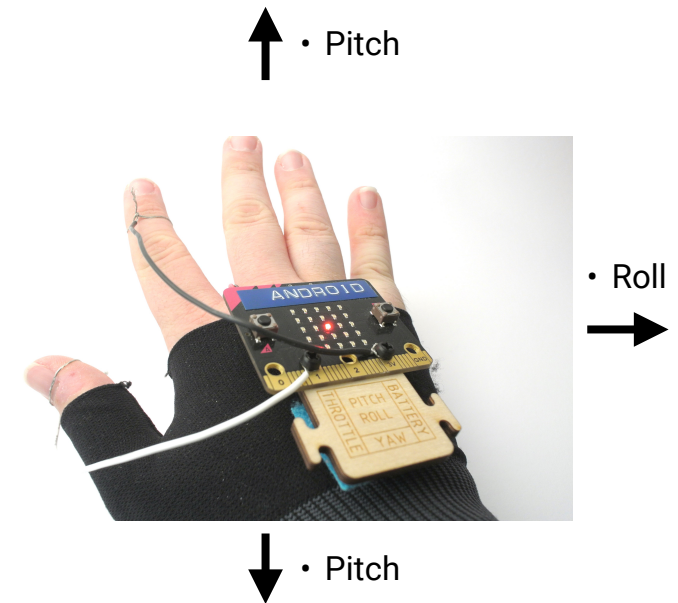
- Skru på Micro:bit ved å koble kabelen til batteriholderen til Micro:biten
- Plasser dronen midt på et åpent område, og skru på strømmen
- Skru på dronen, som normalt
- Trykk på A og B samtidig, for å starte motorene
- For å få mer Throttle, trykker du på B, for å få mindre Throttle, trykker du på A
- Øk gassen (Throttle) til dronen begynner å heve seg, og kommer opp i passe høyde (1-2 meter)
- Juster Throttle til høyden så vidt synker
- Bruk ThrottleBoost, ved å klemme fingersensorene sammen, dronen vil da klatre. Når dronen har klatret nok, slipper du ThrottleBoost, og Throttle vil da gå tilbake til det nivået det var, før du ga en ThrottleBoost



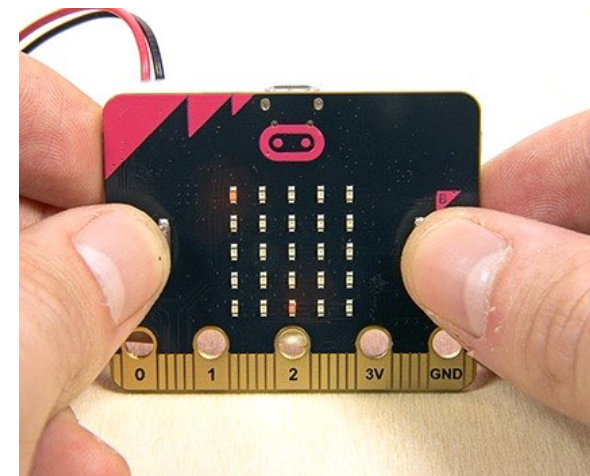
Hvordan bruke hansken

- Du kontrollerer hvilken retning du vil at Air:bit skal gå, ved å gi en hånd-bevegelse, hvor du tilter hånden, i den retningen du vil at Air:bit skal fly
 - F.eks. så vil en hånd-bevegelse rett frem, gjøre at Air:bit flyr rett frem
 - En hånd-bevegelse til venstre, vil gjøre at Air:bit flyr til venstre

• Roll
←

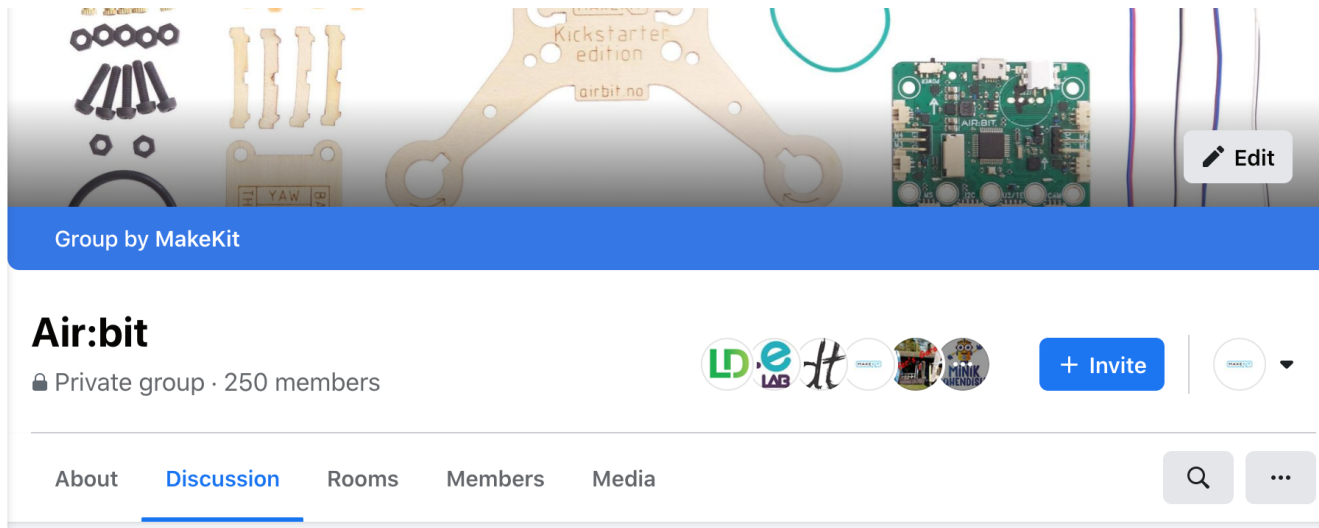


- Du skruv av Air:bit på samme måte som tidligere, ved å trykke på A og B samtidig, eller ved å aktivere panikk-funksjonen ved å riste på håndkontrollen



Kontakt:

Få råd og vink i vår facebook-gruppe
www.facebook.com/groups/goairbit/



www.makekit.no



support@makekit.no



[makekit](#)



[gomakekit](#) (also twitter)